MOTOROLA
Semiconductor Products Inc.

AN-810
Application Note

DUAL 16-BIT PORTS FOR THE MC68000 USING TWO MC6821S

Prepared by:
James McKenzie
Microprocessor Applications Engineering
Austin, Texas

The MC6821 Peripheral Interface Adapter (PIA) is a 40-pin device having two 8-bit ports. Each port has its own control register and may be configured as input or output on a bit-by-bit basis.

Two PIAs may be configured on the MC68000 microprocessor bus to give two 16-bit ports. The ability of the MC68000 to simultaneously access 16 bits of data at an effective rate of up to two megahertz makes it ideal for processing applications using state-of-the-art A/D or D/A converters. The MC68000 is also suited for applications involving advanced peripherals with parallel inputs or outputs and a high data throughput rate.

ASYNCHRONOUS OPERATION

The schematic for the MC68000/MC6821 asynchronous interface appears in Figure 1. Typical timing diagrams appear in Figure 2. Edge connector designations for MC68000 signals correspond to the MEX68KDM Design Module bus pin allocations (EXORciser bus). The asynchronous interface is responsible for three major tasks:

- Detecting when the PIAs are being addressed
- Synchronizing the MC68000 bus cycle to the local E clock
- Controlling data flow to and from the PIAs

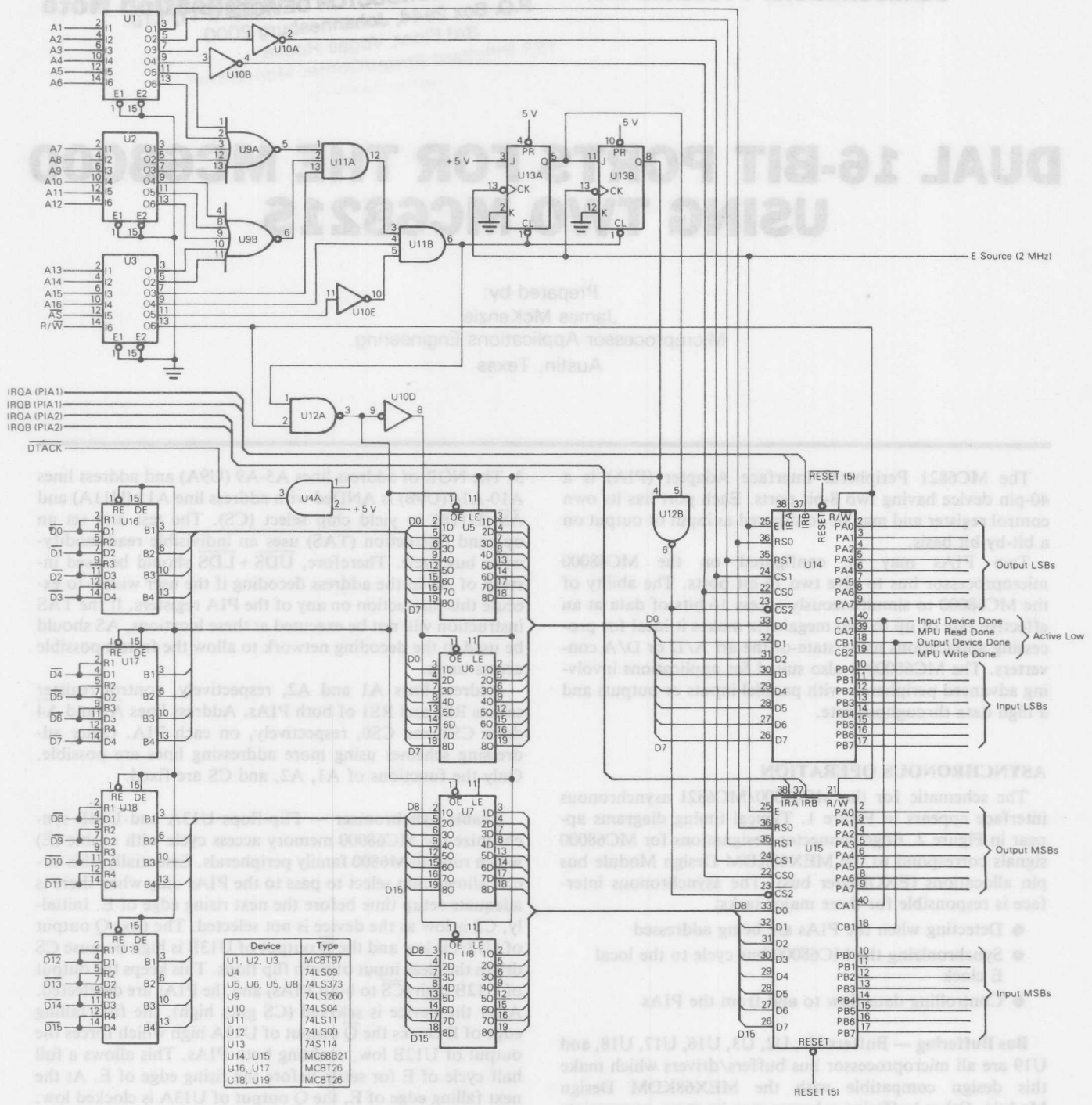
Bus Buffering — Buffers U1, U2, U3, U16, U17, U18, and U19 are all microprocessor bus buffers/drivers which make this design compatible with the MEX68KDM Design Module. Other buffering schemes may be more appropriate in other applications. In particular, buffers U5, U6, U7 and U8 will provide sufficient data bus buffering in a system where the data bus is not inverted.

Address Decoding — The two PIAs are located at $18000-$18007, as shown in the memory map given in Figure 3. The NOR of address lines A5-A9 (U9A) and address lines A10-A14 (U9B) is ANDed with address line A15 (U11A) and $\overline{AS}$ (U11B) to yield chip select (CS). The test and set an operand instruction (TAS) uses an indivisible read-modify-write bus cycle. Therefore, $\overline{UDS} + \overline{LDS}$ should be used instead of $\overline{AS}$ in the address decoding if the user wishes to execute this instruction on any of the PIA registers. If the TAS instruction will not be executed at these locations, AS should be used in the decoding network to allow the fastest possible access times.

Address lines A1 and A2, respectively, control register selects RS0 and RS1 of both PIAs. Address lines $\overline{A3}$ and $\overline{A4}$ drive CS1 and CS0, respectively, on each PIA. Other addressing schemes using more addressing lines are possible. Only the functions of A1, A2, and CS are fixed.

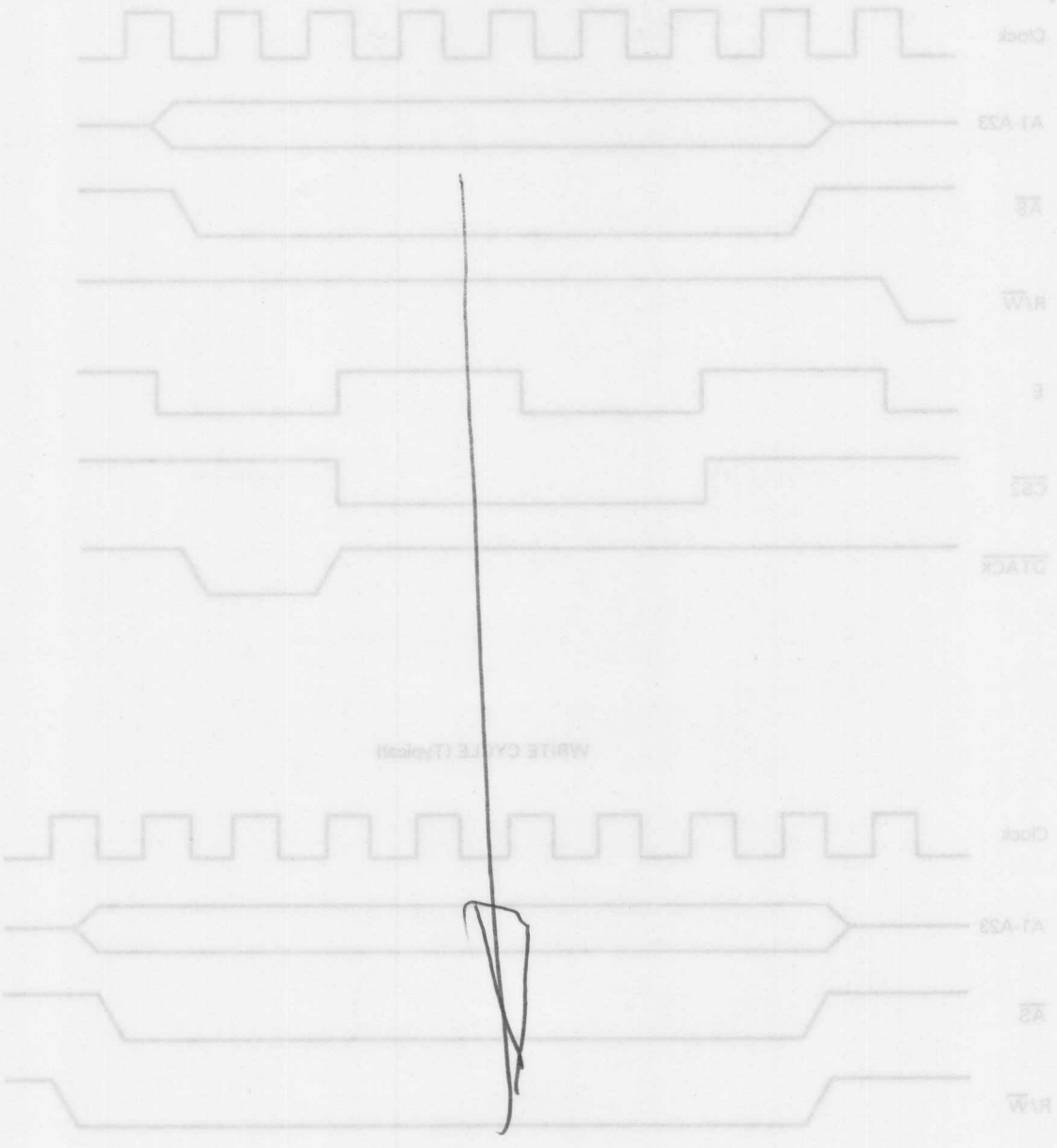
Enable Synchronizer — Flip-flops U13A and U13B synchronize the MC68000 memory access cycle with enable (E) which runs all M6800 family peripherals. Essentially, this circuit allows chip select to pass to the PIAs only when there is adequate setup time before the next rising edge of E. Initially, CS is low as the device is not selected. The next Q output of U13A is low and the $\overline{Q}$ output of U13B is high because CS drives the clear input of both flip flops. This keeps the output of U12B high ($\overline{CS}$ to both PIAs) and the PIAs are deselected. After the device is selected (CS goes high), the first falling edge of E clocks the $\overline{Q}$ output of U13A high which forces the output of U12B low, enabling both PIAs. This allows a full half cycle of E for setup before the rising edge of E. At the next falling edge of E, the Q output of U13A is clocked low. This removes $\overline{CS2}$ from both PIAs (via U12B), latches the data present on the bus (U5, U6, U7, U8) and returns $\overline{DTACK}$ to the MC68000 through an open-collector buffer U4A). This terminates the MC68000 memory access cycle.

If a fifty percent duty cycle, two megahertz E signal is used a best case access cycle time of 875 nanoseconds and a worst case access cycle time of 1375 nanoseconds will be obtained.



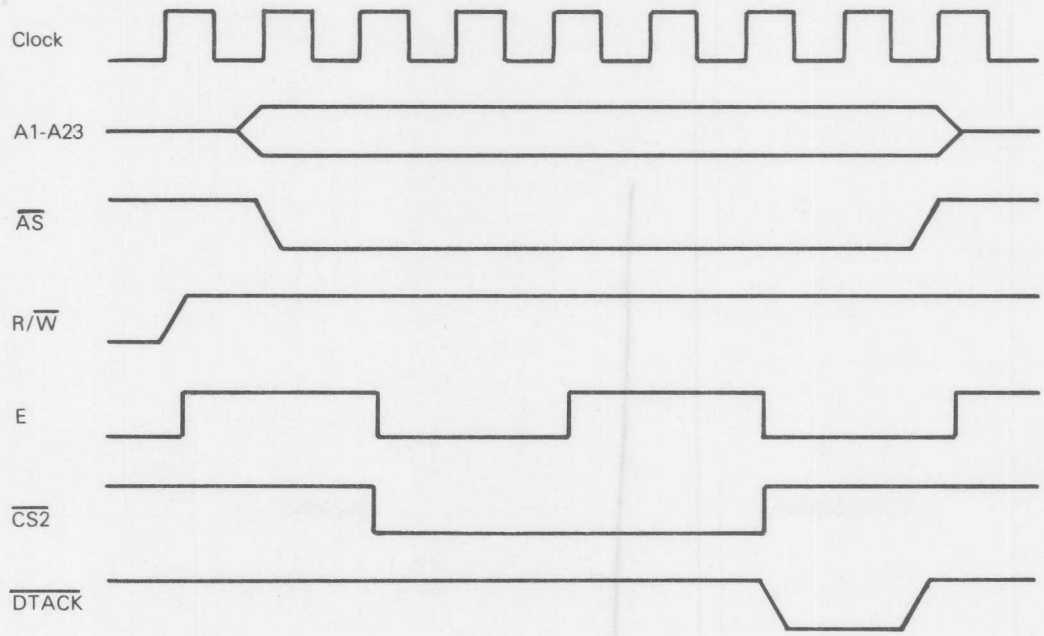
Device	Type
U1, U2, U3	MC8T97
U4	74LS09
U5, U6, U5, U8	74LS373
U9	74LS260
U10	74LS04
U11	74LS11
U12	74LS00
U13	74S114
U14, U15	MC68B21
U16, U17	MC8T26
U18, U19	MC8T26

Figure 1. Asynchronous MC6821 Interface — Schematic Diagram



MOTOROLA *Semiconductor Products Inc.*
Printed in Switzerland

READ CYCLE (Typical)



WRITE CYCLE (Typical)

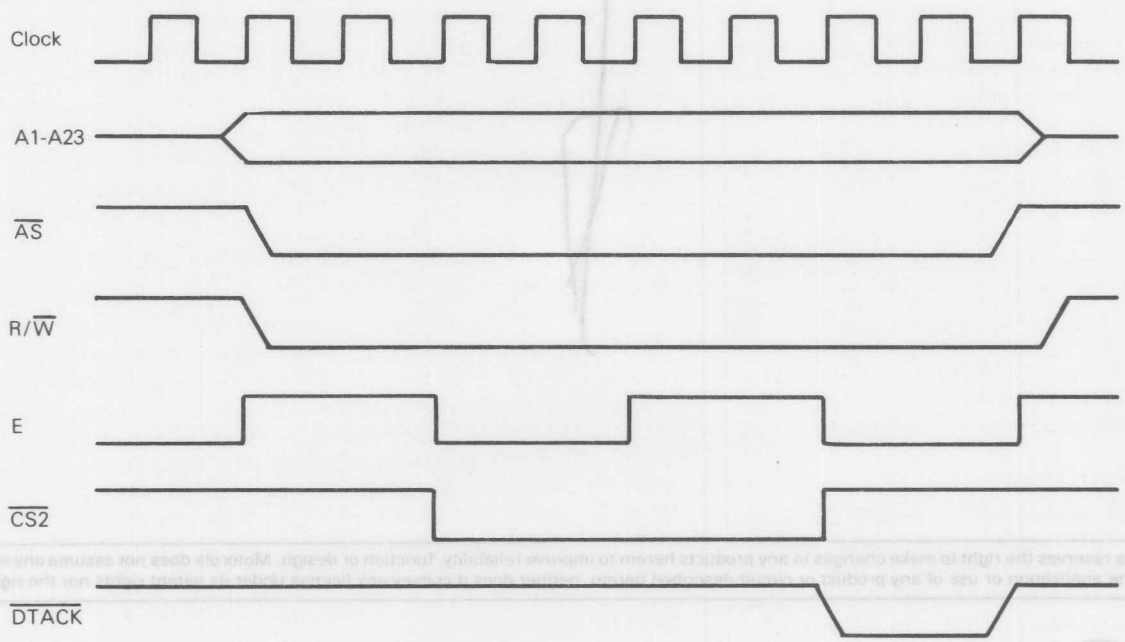


Figure 2. Asynchronous Interface Timing Diagrams

18000 18001	Peripheral Data/DDR, A Side
18002 18003	CRA
18004 18005	Peripheral Data/DDR, B Side
18006 18007	CRB

Figure 3. Memory Map

Bus Buffers Enables — Gates U12A and U10D coordinate the flow of data on the bidirectional data bus to and from the PIAs. These gates form the Boolean equation $\overline{R/W} \bullet \overline{CS}$ (pin 3, U12A) which gates the flow of data to and from the PIAs on U16, U17, U18, and U19. Data is allowed to flow from the PIAs to the MC68000 bus through latches U5 and U7. Also, the signal $R/\overline{W} \bullet \overline{CS}$ (pin 8, U10D) allows data to pass from the MC68000 bus to the PIAs through latches U6 and U8. Latches U5, U6, U7, and U8 guarantee that valid data is on the bus throughout the MC68000 cycle, not just when the PIAs are selected.

SYNCHRONOUS OPERATION

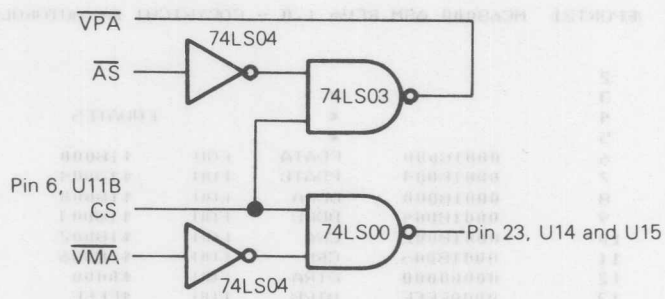
The MC68000 can control M6800 synchronous parts directly using the M6800 peripheral control bus. This bus consists of enable (E), valid memory address ($\overline{VMA}$), and valid peripheral address ($\overline{VPA}$). The two 16-bit ports may be operated synchronously by replacing U13, U12B, and U4A of Figure 1 with the circuitry shown in Figure 4. Valid peripheral address ($\overline{VPA}$), a wire-ORed signal, is returned by an open-collector NAND gate when the chip select is detected. As the processor responds with $\overline{VMA}$, the PIAs are selected (pin 6, U12B). Enable (E) is provided by the MC68000 in this case and has the frequency of the system clock divided by eight. The MC68000 must synchronize $\overline{VMA}$ with E internally, so this method does not allow as fast an access as the asynchronous interface.

SOFTWARE CONSIDERATIONS

Because the upper and lower data strobes ($\overline{UDS}$, $\overline{LDS}$) are not used in address decoding, an individual PIA cannot be accessed. However, word operations on the MC68000 must begin access on an even address. Therefore, the user must take care to address the registers of the PIA with an even address. If individual access is desired, address line A3 could be shifted into the address decoding network and $\overline{LDS}$ applied to CS1 of U14 through an inverter. Likewise, $\overline{UDS}$ could then be applied to CS1 of U15 through an inverter. This would result in the memory map shown in Figure 5.

PERIPHERAL CONTROL LINES

The configuration and labeling of the peripheral control lines (CA1, CA2, CB1, CB2) for the PIAs in Figure 1 is for a 16-bit input port (A sides of each PIA) and a 16-bit output port (B sides) each programmed for handshake operation. Any of the other configurations of these control lines is possible. The most desirable configuration will depend on the type of peripheral equipment being interfaced and its application. A typical initialization routine for the configuration shown in Figure 1 is given in Figure 6.



Note: When $\overline{VMA}$ is used, $\overline{AS}$ should be disconnected from the $\overline{CS}$ decoding (Figure 1, U11B) and that input is tied active.

Figure 4. Synchronous Interface Circuitry

18000	Peripheral Data/DDRA	(U15)
18001	Peripheral Data/DDRA	(U14)
18002	CRA	(U15)
18003	CRA	(U14)
18004	Peripheral Data/DDRB	(U15)
18005	Peripheral Data/DDRB	(U14)
18006	CRB	(U15)
18007	CRB	(U14)

Figure 5. Alternative Memory Map

MODES OF OPERATION

The PIAs may be operated in one of two basic modes, polled or interrupt driven. Polling can cause excessive execution time overhead when more than just a few peripherals are on the bus, so interrupts are usually an attractive alternative. There are many ways to run an interrupt driven system, especially on the MC68000 which has seven priority levels of interrupt and can handle up to 192 unique user interrupt vectors. The MC68000/MC6821 interface yields four interrupt request lines giving a high degree of versatility for interrupting, regardless of the prioritizing scheme used or whether the PIAs are configured as 8-bit ports, 16-bit ports or a combination of both.

32-BIT PORTS

If address line A1 is allowed to drive RS1 and address line A2 drives RS0, then the peripheral data registers for the two PIAs will occupy four consecutive locations of memory beginning at $18000. This location may be used as a 32-bit input or output port. Control register A would be located at $18004 and control register B would be at $18006. Keep in mind that these last two registers are each 16 bits wide, as shown in Figure 7. The 32-bit port could be accessed with long word attribute op codes such as:

MOVE.L $18000,D0

CONCLUSION

Two PIAs provide an excellent parallel I/O port for the MC68000 and are easily interfaced to the standard asynchronous bus. If B series parts are used, the PIAs may be accessed at effective rates of greater than 1.0 megahertz.

```
REPORT21  MC68000 ASM REV= 1.0 - COPYRIGHT BY MOTOROLA 1978          PAGE  1


                          *
                          *                EQUATES
                          *
          00016000  PDATA   EQU     $16000      LOCATION OF PERIPHERAL DATA REGISTER A
          00018004  PDATB   EQU     $18004      LOCATION OF PERIPHERAL DATA REGISTER B
          00018000  DDRA    EQU     $18000      LOCATION OF DATA DIRECTION REGISTER A
          00018004  DDRB    EQU     $18004      LOCATION OF DATA DIRECTION REGISTER B
          00018002  CRA     EQU     $18002      LOCATION OF CONTROL REGISTER A
          00018006  CRB     EQU     $18006      LOCATION OF CONTROL REGISTER B
          00000000  DIRA    EQU     $0000       SETS ALL 16 A LINES AS INPUTS
          0000FFFF  DIRB    EQU     $FFFF       SETS ALL 16 B LINES AS OUTPUTS
          00002525  INITA   EQU     $2525       VALUE TO INITIALIZE CRA
          00002525  INITB   EQU     $2525       VALUE TO INITIALIZE CRB
          00000000  ACDRA   EQU     $0000       VALUE TO ACCESS DDRA THROUGH CRA
          00000040  ACDRB   EQU     $0040       VALUE TO ACCESS DDRB THROUGH CRB
                          *
                          *            INITIALIZE THE PIA'S
                          *
000000 33FC0000
               00018002           MOVE.W  #ACDRA,CRA      OPEN DDRA (16 BITS)
000006 33FC0000
               00018000           MOVE.W  #DIRA,DDRA      A SIDE AS INPUT
000010 33FC2525
               00018002           MOVE.W  #INITA,CRA      HANDSHAKE MODE,INTERRUPT ENABLED
000018 33FC0000
               00018006           MOVE.W  #ACDRB,CRB      OPEN DDRB (16 BITS)
000020 33FCFFFF
               00018004           MOVE.W  #DIRB,DDRB      B SIDE AS OUTPUT
00002B 33FC2525
               00018006           MOVE.W  #INITB,CRB      HANDSHAKE MODE,INTERRUPT ENABLED
                          *
                          *
```

Figure 6. Initialization Routine

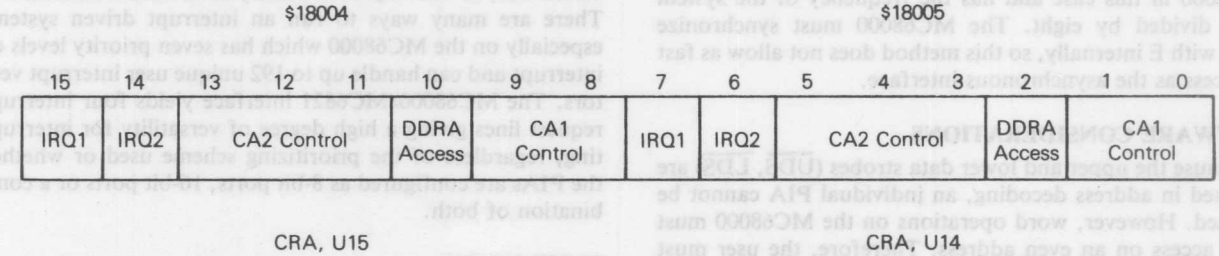


Figure 7. 16-Bit Control Register